Tip of the iceberg

Digital Solution Development

ISoftware

People

People are the cornerstone of digital solutions. They are both the creators and the users of these solutions. While people management skills are crucial, understanding cognitive psychology and how we should engage with people is even more important. This extends to how digital solutions should engage with people.

Cognitive Psychology

The study of mental processes. It explores how people understand, process, and store information. Cognitive psychology also tries to understand how these processes influence behaviour and interactions.

- **Perception**: Prioritise information, minimise distractions.
- **Attention**: Focus on tasks, minimise interruptions.
- **Memory**: Clear communication enhances storage.
- **Language**: Use clear, user-oriented, accessible language.
- **Thinking and Reasoning**: Draw valid conclusions, identify biases.
- **Problem-solving**: Clearly articulate problems, evaluate solutions.
- **Decision-making**: Collect relevant data, assess risks.
- **Learning**: Engage actively, seek feedback.
- **Cognitive Development**: Acquire specialised knowledge, adapt to new situations.

Humane Engagement

How we engage is important. Engagement applies to both the people developing and using the digital solution. Humane engagement is key.

- **Active Listening**: Understand and respond thoughtfully.
- **Empathy**: Understand others' perspectives and feelings.
- **Respect**: Value contributions and feedback.
- **Transparency**: Communicate openly and honestly.
- **Authenticity**: Be genuine and approachable.
- **Positive Communication**: Focus on constructive feedback and appreciation.
- **Inclusivity**: Create a welcoming environment.
- **Collaboration**: Encourage teamwork and open communication.
- **Adaptability**: Be flexible and responsive to change.
- **Recognition**: Acknowledge and appreciate accomplishments.
- **Trust-building**: Build trust through reliable and consistent behaviour.
- **Patience**: Recognise individual abilities and learning paces.
- **Constructive Feedback**: Provide actionable, growth-oriented feedback.
- **Emotional Intelligence**: Manage your own and appreciate others' emotions.
- **Purpose and Meaning**: Connect work to the overall mission.

Process

Every successful digital solution begins with understanding needs. Businesses, like individuals, have needs crucial for survival and growth. Business analysis identifies these needs. In today's world, this change is predominantly digital transformation. Digital transformation is more than digitising existing processes; it's about changing mindsets, reimagining business operations and embracing new technologies. This transformation is where true innovation lies.

Digital solution development is a pipeline that transforms requirements - iteratively and incrementally

1 Business Analysis

Identifying needs that drive change, focusing on increasing revenue or reducing costs. This analysis produces business requirements. This analysis is the engine room of innovation. It also informs what is reasonable when setting goals, objectives, and desired outcomes.

2 Business Requirements

Goals, objectives, and desired outcomes that explain why the change is necessary. They should be **S**pecific, **M**easurable, **A**ttainable, **R**elevant, and **T**ime-bound (SMART). Success is measured by achieving these goals, not delivery. The project ends when the goals are achieved.

3 Stakeholder Requirements

Translating business requirements into stakeholder needs. These are expressed as stories: "As a {stakeholder}, I want {action} so that {value}." Each story should have clear acceptance criteria: {criteria}. Stories should be INVEST. Fulfilling stakeholder requirements ensures business requirements are met.

4 Solution Architecture

Decomposing the solution into components and defining their interfaces. Showing how the components collaborate to deliver the required stakeholder needs. This includes direct, alternate, and exception interactions. An interface is a point of communication or interaction between two components. Observability is crucial for verifying solution correctness and efficiency. Interactions are usually captured as use cases.

User Interface

- Data model
- Visual hierarchy / Appearance
- Interaction
- Empty states
- Data entry validation / Error states

Service Interface

- Request / Response
- Validation
- Error handling

Database Scheme

- Schema design (3rd Normal Form/Document model)
- Data integrity
- Indexing
- Read optimisation

5 Component Requirements

The solution architecture is the solution requirements. The defined interfaces become the requirements for individual components. These are augmented with non-functional requirements, thus ensuring the solution has both the capabilities and qualities required to deliver the expected value.

6 Software Design

The process of converting component requirements into software requirements. An engineering practice applying the following principles.

Cohesion

- Functional
- Communicational
- Sequential
- Procedural
- Temporal
- Logical
- Coincidental

Coupling

- Message
- Data
- Stamp
- Control
- External
- Common

DRY

Don't repeat yourself

SOLID

- Single responsibility
- Open/Closed
- Liskov substitution
- Interface segregation
- Dependency injection

7 Implementation

The software design defines the requirements for the software. It guides implementation and automated testing. The implementation should align with the Agile rules of simplicity:

- All automated tests pass.
- No duplication (code, logic, data).
- Distinct idea or responsibility expressed separately.
- Code is minimal.

Object orientation is primarily a way of thinking before it's a set of concepts in an object-oriented language. Well-designed objects have attributes, operations, and expressive relationships with other objects. This is especially important for classes in the domain model. Avoid anaemic domain models.

Unit testing is essential. It verifies code and serves as a form of documentation. Poorly designed or excessive interfaces can increase the maintenance burden. Fakes are only necessary for external dependencies over which there is no software control.

CI/CD pipelines integrate these tests into an automated workflow, providing rapid feedback on code changes and facilitating continuous improvement.

Product

Successful products fulfil people's true needs. They should give the user what they want, when they want it, and how they want it. Users feel delighted when they are in control and have the freedom to work as they wish. They feel supported when their thinking is understood. Users appreciate being respected and value continuous feedback to improve their experience.

Gestalt Psychology

Understanding visual perception helps designers create groups, establish a visual hierarchy, and direct attention. This is the basis of graphic and UI design.

Figure/ground

Proximity

Similarity

Closure

Continuity

Common fate

Simplicity

Weight

Balance

Usability Heuristics

Rules of thumb which determine if a user interface is usable.

- Visibility of system status.
- Match between system and user worlds.
- Consistency and standards.
- Error prevention.
- Recognition rather than recall.
- Flexibility and efficient use.
- Aesthetic and minimalist design.
- Help recognise, diagnose, and recover from errors.
- Help and documentation.

User Experience

The overall experience of using the digital solution.

Ensure freedom and control, but with good guidance to prevent errors.

If the solution aligns with user expectations, they are likely to achieve goals and find the solution supportive. Mismatch can lead to frustration, unmet goals, and feeling unsupported.

The best way to achieve a good user experience is to ask, engage with, and test it out on users.